REMARKS/ARGUMENTS

Solely in order to expedite prosecution, Claims 8, 17, 21-24 and 28 have been cancelled, and Claims 10, 18 and 25 have been presented in an independent form with minor typographical corrections. As such, it is respectfully submitted that no substantive amendment has been made, and this Amendment should be entered in order to reduce issues. Claims 10-15, 18-20 and 25-27 are pending.

The Examiner provisionally rejected Claims (21-24), (8, 10-12) and (28, 17-19) under obviousness-type double patenting as being unpatentable over Claims (1-5), (8-11) and (18-19) of the co-pending Application No. 09/886,454, respectively. Both applications are still being prosecuted, and upon allowance we will assess filing a terminal disclaimer if appropriate.

The Examiner objected to the abstract and the specification because "Java" is a trademark. A substitute specification is provided which corrects instances of the word "Java" to "Java™" in the abstract and the specification, in accordance with Examiner's suggestion. Accordingly, it is respectfully submitted that the Examiner should withdraw the objections to the abstract and the specification.

The Examiner objected to Claims 8, 11, 14-15, 17-18, 21-23 and 26-28 because "Java" is a trademark. The pending claims have been rewritten to correct the word "Java" to "Java™", in accordance with Examiner's suggestion.

Rejection of claims under 35 U.S.C. 101

The Examiner has rejected Claims 21-27 under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter, and has suggested claims to "a computer-readable medium". Accordingly, Claims 21-24 and 28 have been cancelled, the limitations of Claim 28 have been incorporated into the amended Claim 18 (i.e. claim 18 has been presented in independent form), and currently amended Claims 25-27 are now directed to "a computer-readable medium" and depend from the currently amended independent Claim 18, as recited above.

Rejection of claims under 35 U.S.C. 103

The Examiner rejected Claims 8, 10-12, 15, 17-24 and 27 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,151,703 ("Crelier"). This rejection is traversed below.

(a) Crelier does not teach or suggest: a Java object representation inside a virtual machine having a direct reference to an internal class representation of a class associated with the Java object

Claim 8 has been cancelled and its limitations incorporated into currently amended independent Claim 10. Claim 10 recites a method of identifying Java objects and classes, using object representations having (a) a first direct reference to an internal class representation associated with a Java object, and (b) a second reference to instance fields associated with the Java object.

In contrast, Crelier neither teaches nor suggests a first direct reference to an internal class representation (a). In the Final Office Action, Examiner states that in Crelier Fig. 4, object handle (401) corresponds to the first class reference (a), and cites col. 8:62-64 as teaching a direct reference. However, object handle (401) comprises a conventional method pointer to a method table, with the method table including a list of methods for the objects of the class as well as a pointer to a ClassClass descriptor, wherein the ClassClass descriptor includes information such as class version, class name, and a pointer to a class constant pool. Furthermore, Crelier merely points out that "the method table includes a pointer to the method block for the class", and does not teach a direct reference from the object representation to an internal class representation (Crelier col. 8, lines 62-64). Therefore, Crelier does not teach or suggest a direct reference from an object representation to an internal class representation. Instead, Crelier merely teaches a conventional method pointer to a method table. As noted in the background of the present application, page 3:15-26, which states (page 3:17-28):

> "Typically, there is a reference from the Java object representation directly to a method table that includes the methods associated with the object. Although the direct reference to the method table allows method invocations to be performed, the conventional object representation in Java requires some processing to find information about the object (e.g., object type, object size, static fields, etc.) Such information about the Java object can be stored in the internal class representation of the object. In other words, the virtual machine typically internally represents and stores the information associated with the

Java object's class. However, accessing this information takes up valuable processing time. This can seriously hinder performance of virtual machines, especially in systems with limited computing power and/or memory." (Specification, page 3, lines 17-28)

(b) Crelier does not teach or suggest: (1) sequential reading by a virtual machine at runtime of a cluster of object representations or (2) marking of classes using a direct class reference

Claim 10 recites a method of identifying Java objects and classes, comprising sequentially reading a cluster of object representations, determining whether objects or classes are to be identified, and using the object representations to mark objects or classes in memory.

In the Final Office Action, the Examiner states that Crelier teaches: (1) sequential reading of object representations (at cols. 8:13-14 and 8:19 20). However, Crelier merely states that "handles for the various objects are maintained in a global array for facilitating object processing, such as garbage collection." Also, the Examiner has asserted that Crelier teaches: (2) marking classes using a direct reference to a class (col. 8:42). However, Crelier merely teaches an "unsigned long totalhash" as part of a ClassClass struct.

Again it should be noted that the object handles (401) in Crelier are conventional object handles and do not teach a direct reference to an internal class representation in a virtual machine. Hence Crelier cannot possibly teach or suggest (1) sequential reading by a virtual machine at runtime of a cluster of object representations or (2) marking of classes using a direct class reference. Crelier does not permit access to (and subsequent marking of) classes in memory except by following indirect paths from an object handle to information associated with the object's class.

Also marking both objects and classes using a cluster of object representations having direct class references is not taught or suggested by Crelier. The advantage of a cluster is pointed out in the background of the present application, which states:

"Furthermore, using conventional Java object representations, it is difficult to implement a single "garbage collection" scheme that would allow removal of Java objects, as well as Java classes. In other words, conventionally, one garbage collection method is used to remove Java objects when they are no longer needed, and another garbage collection method is used to remove classes from memory when there are no longer needed. Thus, conventionally, garbage collection can use a significant amount of memory and

computing time of conventional virtual machines." (Specification, page 3, line 29 to page 4, line 3)

(c) Crelier does not suggest a cluster of object representations inside a virtual machine

In the Final Office Action, the Examiner has stated that while Crelier does not explicitly disclose a cluster, it would have been obvious to modify Crelier's teaching "to expand from one array to more than one array [...] because one of ordinary skill in the art would have been motivated to make the system more robust." (Final Office Action, page 16)

However, it is respectfully submitted that the general concept of "robustness" does NOT suggest a cluster of object representations. As such, the Examiner has failed to provide a motivation or suggestion in the cited art itself or general knowledge for modifying Crelier. In fact, as noted above, Crelier fails to even teach a single object representation consisting of a first and second reference in the context of the claimed invention. Accordingly, it is respectfully submitted that Crelier cannot be possibly modified to teach the claimed invention.

For at least the above reasons, Applicants respectfully submit that Claim 10 and dependent Claims 11-15 are allowable over the cited prior art.

Claim 28 has been cancelled and its limitations incorporated into currently amended independent Claim 18. Claim 18 recites a computer-readable medium embodiment of the method recited in Claim 10, and should be allowable for the same reasons outlined above with respect to Claim 10. Therefore, Applicants respectfully submit that Claim 18 and dependent Claims 19-20 and 25-27 are allowable over the cited prior art.

Conclusion

Based on the foregoing, it is submitted that Claims 10-15, 18-20 and 25-27 are patentably distinct over the cited art of record. Additional limitations recited in the independent claims or the dependent claims are not further discussed because the limitations discussed above are sufficient to distinguish the claimed invention from the cited art. Accordingly, Applicants believe that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner.

Applicant hereby petitions for an extension of time which may be required to maintain the pendency of this case, and any required fee for such extension or any further fee required in connection with the filing of this Amendment is to be charged to Deposit Account No. 500388 (Order No. SUN1P832). Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP

Behfar Bastani-Booshehri
Reg. No. 52,599

P.O. Box 70250
Oakland, CA 94612-0250
(650) 961-8300